EXHIBIT 16

If you'd like to run your own instance of this server, you'll need to deploy it somewhere and configure it with your API keys. A few easy options are provided below, though you can also deploy it to any other service you'd like if you know what you're doing and the service supports Node.js.

Self-hosting

See here for instructions on how to self-host the application on your own VPS or local machine.

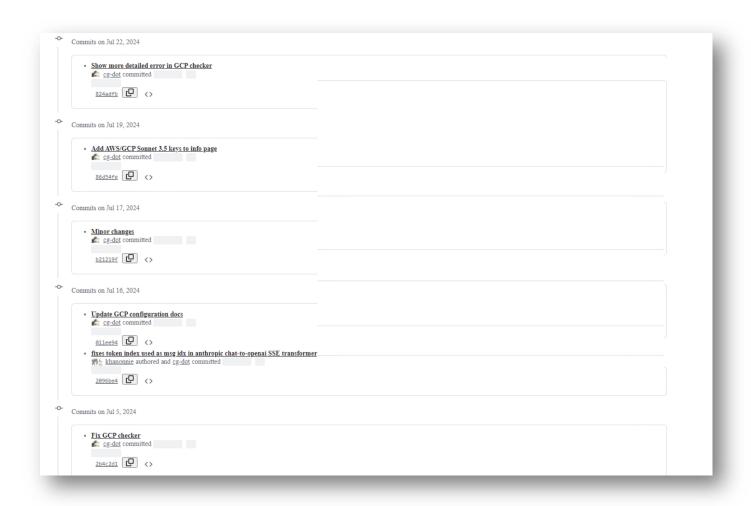
Ensure you set the TRUSTED_PROXIES environment variable according to your deployment. Refer to .env.example and config.ts for more information.

Fiz and Sekrit are working on some alternative ways to deploy this conveniently. While I'm not involved in this effort beyond providing technical advice regarding my code, I'll link to their work here for convenience: Sekrit's rentry

P Huggingface (outdated, not advised)

See here for instructions on how to deploy to a Huggingface Space.

See here for instructions on how to deploy to Render.com.



```
146 + # For GCP credentials, separate the project ID, client email, region, and private key with a colon.
145
                 146
          148
                 147
                # See 'docs/aws-configuration.md' for more information, there may be additional steps required to set up AWS.
148
          150
                  \textbf{ANS\_CREDENTIALS} = \textbf{myaccesskey} \texttt{id:mysecretkey:us-east-1,anotheraccesskey} \texttt{id:anothersecretkey:us-west-2}
149
          151
                 # See 'docs/azure-configuration.md' for more information, there may be additional steps required to set up Azure.
150
          152
                  \label{eq:advector}  \textbf{AZURE\_CREDENTIALS} = \texttt{azure-resource-name:deployment-id:api-key,another-azure-resource-name:another-deployment-id:another-api-key} 
          153 + GCP_CREDENTIALS=project-id:client-email:region:private-key
```